Continuous, transparent network access for portable users

A Mobile Networking System based on Internet Protocol

CHARLES E. PERKINS, PRAVIN BHAGWAT

In the last two years, we have witnessed two major changes in computer technology. First, portable computers as powerful as some desktop workstations in terms of computing power, memory, display, and disk storage are beginning to appear. Second, with the availability of wireless adapter cards, users of laptop computers are no longer required to remain confined within the wired LAN premises to get network access. Users of portable computers would like to carry their laptops with them whenever they move from one place to another and yet maintain transparent network access through the wireless link. By transparent network access we mean the ability of a mobile user to set up and maintain network connectivity despite migration from one network to another. This movement may possibly introduce a momentary pause in the operation, but it should not require reinitialization of network sessions. The existing set of network protocols do not meet this requirement since they were designed under the assumption of a stationary network topology in which hosts do not change their location over time.

The problem of providing continuous network connectivity to mobile computers has received considerable attention [1-4], especially in the context of networks based on the Transmission Control Protocol/Internet Protocol (TCP/IP) [5, 6] suite of protocols. The proposed solutions either require changes to the existing network architecture [3] or introduce new encapsulation protocols [1, 4] to handle this problem. In contrast, our approach, which is based on the use of a natural model and an existing IP option, does not introduce any new protocol and achieves optimal routing [7, 8]. The solution is transparent to transport and higher layers, and does not require any changes to stationary hosts and routers.

Our model is natural, because we coordinate a collection of mobile hosts (MHs) as a new IP network. As with any IP network, we route packets to the MHs by using a router. Our router is special because once it receives a packet, it does special things to ensure its safe delivery to its destination (the MH). However, this special operation is invisible to existing hosts and routers, so all the routing difference due to movement of the hosts can be hidden and effected by mechanisms under the control of our special entities. The other part of our model, which is a very natural part of a physical wireless data communications system, is the transceiver (access point), which collects wireless packets from a MH for delivery to existing hosts along existing wired networks. This transceiver is required for wireless communications, and it provides the reference point by which the location of the MH is known.

We have implemented our scheme on a set of IBM PS/2 Model 80s running AIX version 1.2. In this article, we present an overview of our scheme and provide some details of our current implementation. The details provided in this article along with [7, 8] can serve as a guide for interested readers who would like to add mobile networking features to their network testbeds.

The Mobility Problem

The Internet is a large collection of networks that share the same address space and inter-operate using a common set of protocols, specifically IP and TCP [5, 6], but including numerous others. It is desirable that the integration of mobile computers within the existing Internet be completely transparent to the transport and higher layers so that users of mobile computers can continue to run existing applications. Any acceptable solution for mobility should interoperate with the existing infrastructure and not require any modifications to existing host or router software. However, this goal is not easy to achieve in practice. The way in which the goal may be met depends in large part upon the precise nature of the assumptions made about existing hosts and protocol implementations.

An Internet address can be thought of as consisting of two parts, the network identifier and the host identifier. All hosts residing on a (sub)net are required to have the same (sub)net address. Within a (sub)net, all attached hosts have a unique host ID. The routing infrastructure uses the network part of the address to route the packet to the correct network. Historically, an Internet address served the purpose of a unique host identifier, but the location information was also effectively embedded in it. When a host moved to a new network, it would acquire a new address. Since the transport layer and network applications assume that network addresses do not change during the lifetime of a connection, the dynamic assignment of new addresses cannot be done without affecting them. To provide application transparency, it is desirable to devise a method by which hosts retain their home addresses and continue to receive packets despite their migration from one network to another.

1070-9916/94/$04.00 © 1994 IEEE

Over the last two years several proposals have been made to address this problem [1, 3, 4]. The scheme proposed by Ioannidis [1, 9] relies on a group of cooperating mobile support routers (MSRs), which advertise reachability to the same (sub)net. Each MH, regardless of its location within a campus, is always reachable via one of the MSRs. When a host sends a packet to a MH, it first gets delivered to the MSR closest to the source host. This MSR encapsulates the packet and delivers it to the target MSR, which strips the encapsulation header and relays the original packet to the MH. This approach is optimized to work within a campus environment and requires additional features before it can be extended to support wide-area mobility.

In Sony's proposal [3], a MH is assigned a new temporary address when it is attached to a new network. The mapping between the home address and the temporary address of a MH is kept in an address mapping table (AMT), which is maintained at the routers. Packets transmitted to the home address of the MH get intercepted by some router that holds an AMT entry for the MH. An address conversion is performed by the router before the packets are forwarded to the physical location of the MH. This method requires modifications to routers and host software and has problems interoperating with the existing hosts unless so-called conversion gateways are used.

Another proposal to support MHs is from Matsushita [4]. This method is also based on the encapsulation approach. A MH is assigned a temporary address when it visits a new network. The packets destined to the home address of the MH are intercepted by a packet-forwarding server (PFS). The PFS encapsulates the packet and forwards it using the temporary address of the target MH. The problem with this method is that routing is always suboptimal unless the software on all stationary hosts is modified.

We have abstracted out particular functions necessary for a mobile networking solution and built our system to use just those functions. Identifying the minimal set of features allowed us to work toward a solution with few encumbrances stemming from our model. Our model was naturally suggested by the idea of segregating the MHs into their own distinct network. This new network is a logical or mobile network, not a network corresponding to a particular extent of wire. Once we decided to use this model, and thus agreed to create a router for the mobile network, we only needed to design the way packets are delivered from the router to the MH. This was done naturally enough by designing the mechanism for packets to find their way to the current location of the MH as defined by its connection to its current access point. Since any wireless MH has to have a transceiver to connect up with, we already had the last necessary functional piece of our model and then set about the task of making the necessary changes to the network protocol implementation in the access points and the single other agent, which is the router for our new network.

Our approach [7, 8] is based on the use of an existing IP option and therefore does not require any changes to the existing hosts and routers. The key idea is that each packet originating from a MH contains enough routing information to be used by the remote host to send a reply back to the source along an optimal path. In the rest of this article, we first present an overview of our scheme and then describe our implementation.

The System

In this section, we will describe the basic entities in our system and point out why the problem solution becomes more natural when the model includes the appropriate entities, thus making implementation and administration of the overall system easier.

Model Definitions and Assumptions

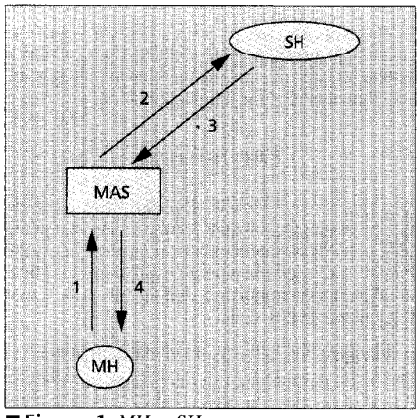
Our system involves the participation of three types of entities, the MH, mobile access station (MAS), and mobile router (MR). The networking architecture that we assume is that of a set of MASs connected through a wired backbone. An MAS supports at least one wireless interface and functions as a gateway between the wired and wireless sides of the network. Due to the limited range of wireless transceivers, a MH can set up a direct link-layer connection with an MAS only within a limited geographical region around it. This region is referred to as an MAS's cell. The geographical area covered by a cell is a function of the medium used for wireless communication. The range of infrared cells is typically limited to about 20 ft, while that of radio frequency cells could be significantly larger.

Any acceptable solution for mobility should interoperate with the existing infrastructure and not require any modifications to existing host or router software. However, this goal is not easy to achieve in practice.

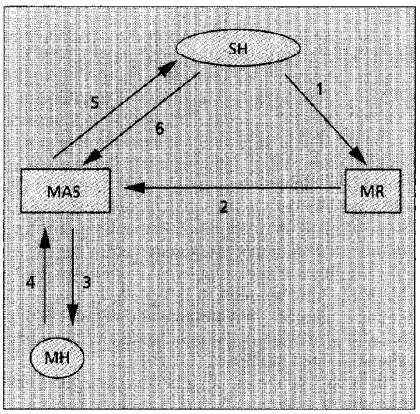
Within one campus or administrative domain there could be multiple (sub)networks reserved for MHs. Each (sub)network has a separate router, the MR. Unlike other routers, an MR is not required to have an interface corresponding to the wireless (sub)net it serves. If an MR has a wireless interface, it can also function as an MAS. The association between an MH and its current MAS is kept in a location directory (LD) maintained at the MR.

An MH retains its address regardless of the MAS cell it is in. It can start sessions with other hosts (both mobile and stationary) and move into other MAS cells without disrupting any active sessions. The movement of an MH is completely transparent to the running applications, except possibly for a momentary pause that may occur while the cell switch takes place. An MH can reside in the cell of only one MAS at any given time. Even if cells of two MASs spatially overlap, an MH routes its outgoing packets through only one of them. An MAS can have multiple MHs in its cell.

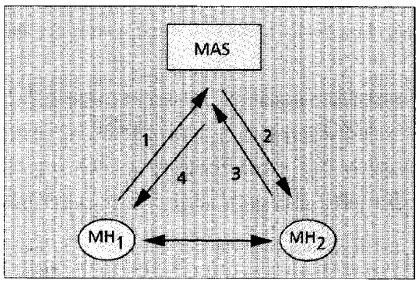
We use the term "correspondent host" (CH) to refer to the host communicating with an MH. In the following discussion, a stationary correspondent host is also referred to as a "stationary host" (SH).



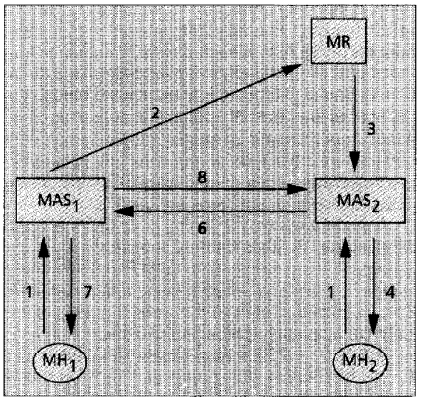
■ **Figure 1.** *MH to SH.*



■ **Figure 2.** *SH to MH.*



■ **Figure 3.** *MH to MH (same cell).*



■ **Figure 4.** *MH to MH (different cell).*

Implications of Our Model

As a result of the way in which we have framed the problem, solutions occur more naturally within the visible design space. We basically propose that the movement of MHs can be enabled by solving a simply stated routing problem. Namely, we can achieve our goals by finding a way to route packets between the MR and MAS.

Since the MHs are considered to be on their own network, we can provide for ever-larger numbers of MHs by adding more mobile networks. When an MR, for capacity reasons, cannot route any more packets to the MHs on its networks, a new MR can be placed into service for new mobile networks. There does not have to be any special relationship between the MRs for different mobile networks. Likewise, there need not be any relationship between the MRs and any MASs. As we will indicate, the MAS can deliver packets to any MH within its cell regardless of what mobile network that host resides on. In fact, in our design, the MASs are fairly passive devices, existing only to serve whichever MHs come its way, and needing coordination with no other MASs in the system or any other entity. Thus, one can hope for an eventual implementation of MASs that costs very little in hardware — perhaps looking like a smoke detector mounted somewhere within a room.

As a result of the lack of interdependence between the model entities, our system is very easy to administer. MASs, MRs, and certainly MHs can be added as needed. It is also possible to allow MHs to receive dynamically assigned IP addresses, which would eliminate even the last usual requirement for operation within an IP network. The means by which an IP address is allocated would also have to allow the MH to know which mobile network it is residing on.

Control Functions

Aside from the routing functions just mentioned, we have some other simply stated functions that are required. First, a MH needs to know when it has entered a cell. We model this process of cell discovery as a client/server interaction, with the MAS advertising its service just by announcing its IP address for use by the MH. The MH accepts service just by picking out one of the servers (MASs) in its area to send packets through.

A location update function is also needed to allow the MR to know where all of its MHs are. We model this as another simple client/server transaction, this time again with the MH as client. The client contacts its location directory server (conveniently located at the MR) with new updates as necessary — that is, whenever the MH decides to accept service from another MAS. Thus, the MH is largely in control of its fate. The MASs interact neither with the MH nor each other, and do not provide any handoff services.

Last, we provide a simplistic method by which

the MASs know which MHs are in their cells. This could be done by a link-level interaction, but for our purposes it has been sufficient to allow the MR to provide location updates to whichever MASs are affected whenever an MH moves. If this information was not made available, an MAS would mistakenly transmit a packet into its wireless range whenever it received one, even if the desired MH had moved out to a new cell at some time in the past. We prefer instead for the MAS to forward the packet back to the MR for further processing.

Simplest Possible Operation

Suppose for the moment that we had a working method by which packets could be delivered from an MR to the current location (MAS) of an MH. Suppose also that the control functions mentioned above were operational. Then, with just these few elements, we can already build a working system to allow MHs free movement between MAS cells. The problem is that all packets destined for the MHs have to go out of their way, because they need to visit the MR before they can be delivered to the current MAS. Packets from an MH to an existing host (CH) do not experience this problem. Thus, this routing phenomenon is called "triangle routing," with the MR, MAS, and CH labeling the vertices of the triangle. Thus, if we devise a method for delivering packets between the MR and the current MAS, our remaining problem will be to find ways of eliminating this triangle routing, which can represent a big loss of routing efficiency. It turns out that this is possible to do in a fairly elegant way, as long as data packets can carry along current routing information with them.

Overview of the Scheme

Our scheme is based on the use of IP's loose source route (LSR) option. The LSR option provides a means for the source host to supply partial routing information to be used by routers in forwarding the datagram to the destination. A source can specify a list of routers to be visited in the specified sequence before the datagram is delivered to the final destination. According to the host requirements document [10], return traffic to the originator of the LSRed packet is also sent with the LSR option by reversing the route taken by the incoming packets. We use this technique to achieve optimal routing between an MH and a CH. There are four possible communication scenarios depending on whether the CH is stationary or mobile and, if the CH is mobile, whether or not the MH and the CH are in the same cell. We consider each case separately and show how optimal routes are constructed in each scenario.

Mobile Host to Stationary Host

An MH, while communicating with an SH, issues packets with the LSR option which specifies that packets should be routed via the MAS serving the MH (arcs 1 and 2, Fig. 1). The SH sends reply packets with the LSR option containing the reversed route. These packets are first delivered to the MAS, which forwards them to the MH. Notice that if the LSR option is not used in the reply packets, these packets will get delivered to the router (MR) for the MH's (sub)network (subsequently called the wireless subnet). The MR would eventually forward these packets to the MH; however, the complete path followed by the reply packets would not be optimal in this case.

Stationary Host to Mobile Host

An SH need not be aware of the current location of the MH when it initiates a session. If it is not aware, the packet sent from the SH (arc 1, Fig. 2) arrives at the MR, which advertizes reachability to the wireless subnet. The MR, using the information in its LD, inserts the LSR option in this packet, which causes this packet to be delivered to the MH via the current MAS serving the MH (arcs 2 and 3, Fig. 2).

Our scheme is based on the use of IP's loose source route (LSR) option. The LSR option provides a means for the source host to supply partial routing information to be used by routers in forwarding the datagram to the destination.

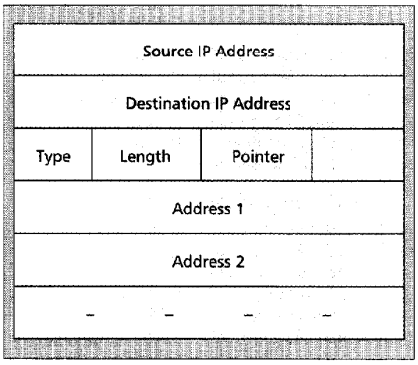
When a reply to this packet is sent, the MH reverses the LSR and sends the packet back to the SH via the MAS (arcs 4 and 5, Fig. 2). Once the SH receives a source-routed packet, it can send subsequent packets to the MH along the optimal path by reversing the incoming LSR.

Mobile Host to Mobile Host Within the Same Cell

An MH does not keep track of other MHs residing in the current MAS's cell. It always uses the current MAS as its default gateway for all outgoing traffic. When an MH initiates a session with another MH, it sends all packets to the MAS just as it would do if it were to send those packets to an SH. Since the MAS keeps a list of all MHs residing in its cell, it can forward those packets to the destination MH. If the wireless link-layer technology supports direct MH-to-MH communication, the MAS can also send an ICMP [11] redirect message to the source MH so that it can directly communicate to the destination MH rather than source-routing its traffic through the MAS. Figure 3 illustrates MH-to-MH communication within the same cell.

Mobile Host to Mobile Host in Different Cells

An MH does not inspect the destination IP address to determine whether the destination host is an SH or MH. Consequently, it always starts off by sending packets with the LSR option. By normal routing mechanisms, these packets are forwarded to the MR associated with the destination MH (arcs 1 and 2, Fig. 4). The MR extends the existing LSR option by inserting the address of the the MAS presently serving the destination MH, and then forwards the packet. Normal routing procedure ensures that these packets get delivered to the



■ **Figure 5.** *An IP header with the LSR option.*

MAS serving the destination MH, followed by the destination MH (arcs 3 and 4, Fig. 4). Notice that the LSR option list of the incoming packet contains the addresses of two MASs, one serving the source MH and one the destination MH. The reply packets are sent by reversing the incoming LSR, which follows the optimal path (arcs 5, 6, and 7, Fig. 4). Once the source MH receives a packet back from the destination MH, it can also send the subsequent packets along the optimal path.

Implementation

We have implemented the aforementioned scheme on a set of IBM PS/2 Model 80s running AIX version 1.2. Each of these machines is equipped with an infrared (IR) wireless adapter card supporting a data transfer rate up to 1 Mb/s. The range of an IR transceiver is limited to about 20 ft. This adapter card uses an Ethernet chip. From the perspective of the device driver, a wireless interface behaves much like an Ethernet interface, since both use a carrier sense multiple access (CSMA) protocol. The only difference is that the wireless adapter card does not support collision detection (CD). This shortcoming, however, did not affect us much because most of our experiments were limited to a small cell population.

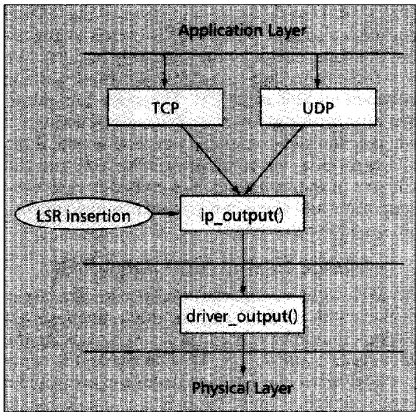
The basic idea of IP's LSR option is to enable any data packet to include routing information so that a particular packet would follow a routing path possibly different than the path taken by normal data packets. This is done by including, in IP's option data fields, the necessary addresses of the desired intermediate routers, along with some ancillary fields to manage the consumption of the routing data (Fig. 5). These fields indicate first, the number of intermediate routing nodes, and second, the next desired intermediate router in the list. When one of the intermediate nodes is reached, the next intermediate routing point is taken out of the list and placed in the destination field of the IP header; then the intermediate router copies the IP address of its own outgoing interface into the IP options data, and finally increments the pointer past its own address. This causes the final destination to have a natural reverse path through the intermediate routing agents.

There are actually two different kinds of IP source-routing options. We use LSR because we only want to include the relevant MAS addresses in our source routes. The other variety of IP source routing is called "strict source route." When using this option, every intermediate routing node must be included in the IP option data. These two source-routing options are distinguished by the use of different IP option numbers.

Our existing implementation consists of approximately 800 lines of kernel code and 1500 lines of user code. It can be thought of as consisting of two parts, the packet routing part and the location information management part. Actions related to packet routing are performed in the kernel. To avoid creating new data structures, location information is stored implicitly in the kernel routing table. This approach has some obvious advantages. First, minimal kernel modifications are needed to route packets to/from MHs. Secondly, with a little modification, the existing route command can be used to manipulate the location information. In the following sections, we first describe how packets are routed among various components and how location information is managed, and then outline the processing required at each component.

Packet Routing

For each MH that has an address on the wireless (sub)net served by an MR, a host route is maintained by the MR. The current location information of the MH (i.e., the address of the MAS serving the MH) is kept in the gateway field of the routing table entry. This routing table entry is distinguished from other entries by the presence of a new flag called RTF_MOBILE. Since the MR advertizes reachability to the range of addresses on the mobile (sub)net, an IP packet destined to an MH is first routed to the MR for further delivery. At the MR, one of the host routes with an RTF_MOBILE flag is chosen to route this packet. The MR knows how to interpret the RTF_MOBILE flag, which specifies that the MR should insert the LSR option in this packet (by using the MAS as the intermediate hop) before forwarding it. Due to the inserted LSR option, this packet is delivered to the MAS currently serving the destination MH. The LSR option is processed here, and, finally, the packet is delivered to the MH.



■ **Figure 6.** *Kernel processing at the MH.*

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.